Seismic Signal Classification using Multi-Layer Perceptron Neural Network

El Hassan AitLaasri
LETSMP, Faculty of science,IbnZohr University, Agadir, Morocco

Es-SaïdAkhouayri
LETSMP, Faculty of science, IbnZohr University, Agadir, Morocco

DrissAgliz
LETSMP, Faculty of science, IbnZohr University, Agadir, Morocco

AbderrahmanAtmani
LETSMP, Faculty of science, IbnZohr University, Agadir, Morocco

ABSTRACT
The aim of the present study is to investigate and explore the capability of the multilayer perceptron neural network to classify seismic signals recorded by the local seismic network of Agadir (Morocco). The problem is divided into two main steps, the feature extraction step and classification step. In the former, relevant discriminant features are extracted from the seismic signal based on the time and frequency domains. These are selected based on the analysts' experience. In the latter step, a process of trial an error was carried out to find the best neural network architecture. Classification results on a data set of 343 seismic signals have demonstrated that the accuracy of the proposed classier can achieve more than 94%.

General Terms
Pattern Recognition, Classification.

Keywords
Seismic signal, classification, multilayer perceptron neural network, feature extraction.

1. INTRODUCTION
Seismic waves can be produced by many types of sources. The latter include tectonic, quarry blast, underground nuclear explosions and cultural activities. These seismic waves are detected by seismic monitoring networks and then converted to digital signal and stored. The recorded signal is then treated and analyzed to extract essential information. The fundamental first step in seismic signal processing is to assign each detected event to a class representing the type of physical phenomena generating it. Such operation leads to a classified seismic database, which can be used in future studies. In view of the very high volume of data, the work conducted by analysts to identify the source of each detected event is considerable and tedious. Therefore, constructing a reliable automatic system for recognizing the seismogram of each incoming event is crucial. This problem has been faced by seismologists in all over the word since the introduction of digital seismic monitoring systems. A review of the literature shows that many different classification methods have been proposed. These methods incorporate spectral ratio of seismic phases [1]-[6], statistical analysis [7][8],cross-correlation [10], and wavelet analysis [11]. Limitation of the deterministic models have led to more sophisticated techniques that integrate artificial neural networks to implement the classifier (e.g., Falsaperla et al., 1996 [12]; Musil and Plesinger, 1996 [13]; Muller et al., 1999 [14]; Dowla et al., 1990 [15]; Tiira, 1999 [16]; Jenkins and Sereno, 2001[17]; Ursino et al., 2001[18]; Del Pezzo et al., 2003[19]; Scarpetta et al., 2005 [20]; Yıldırım et al, 2010 [21]).

Recently, artificial neural networks have attracted increasing attentions for solving many real word problems where traditional techniques cannot be used or are found to be insufficient to describe and model the behavior of the problem. The power of neural networks lies in their ability to model extremely complex non-linear mappings, and in their capability to learn easily the input/output relationship directly from the data being modeled. On the contrary, the most traditional methods require a good understanding of the problem.

The aim of the present study is to investigate and explore the capability of the multilayer perceptron (MLP) neural network to classify seismic signals recorded by the local seismic network of Agadir (Morocco). This problem is divided into two main steps: the first is the processing and feature extraction step, where the relevant characteristics of the signal are extracted. Six features, regularly used by analysts, were considered. The second is the classification step, where a process of trial an error was carried out to find the best MLP architecture (the number of hidden neuron, activation function …etc) and training algorithm. To do so, many configurations were tested.

Four classes are considered in this study. These are: noise (NS), local earthquake (LE), regional earthquake (RE) and quarry blast (QB).

The rest of this paper is structured as follows. The second section gives a brief explanation of the features used. The third section describes the functioning of the MLP neural network classifier. The fourth section presents classification results and illustrates the performance of the classier. Finally, the fifth section reports some conclusions.

2. DATA AND FEATURE EXTRACTION
The data used in this work are collected by the local seismic network of Agadir. The latter consists of five stations deployed around Agadir city (Morocco). Each station consists of a vertical-component short-period seismometer with an output proportional to ground velocity. Seismic signals are continuously acquired and transmitted in real-time via a terrestrial phone line to the national database in Rabat city, and via a radio-frequency FM modulated to the local data center in Agadir city, where they are digitized and analyzed (Figure 1).Each detected event is recorded with the pre-event and post-event data in order to assure complete recording of seismic events. The employed detector is a power-based time domain trigger, whereby the power within a long time-

window (LTA) is continuously contrasted with the power within its consecutive short time-window STA [22]. The algorithm alerts detection and triggers the recording process whenever the STA/LTA ratio exceeds a pre-defined threshold. In this manner, diverse event types are continually detected. In addition to earthquake events, numerous quarry blast seismograms are recorded on a daily basis. This is due to many quarries located surrounding Agadir city. Many other diverse seismic events are also detected. These are often generated by seismic sources such as wind, ocean waves and cultural activities (e.g., machinery). Typical vertical-component seismograms of four different seismic events are plotted in figure 2.

2.1 Feature extraction

The goal behind the feature extraction step is to reduce the high dimentionality of each seismogram to a limited and relevant set of signal characteristics that are able to represent the maximum information existed in the seismogram. In this work, the feature set mostly used by seismic signal analysts corresponds to:

Envelop

The envelope similarity is a measure of similarity between the signal shape e of each incoming event and the reference shape e^r determined from explosion events. The envelop similarity E_s is measured using Manhattan distance:

$$E_s = \frac{\sum_{i=1}^{N} |e(i) - e^r(i)|}{\sum_{i=1}^{N} e(i)}$$

where

$$e(i) = \sqrt{z(i)^2 + HT[z(i)]^2}$$

z(t) is vertical component seismogram, and HT indicates the Hilbert Transform.

Duration

The duration T_d is defined as the total duration in seconds of the event record from the P wave onset t_p to the end of the signal t_{end} defined as the point where the signal is no longer seen above the noise.

$$T_d = t_{end} - t_p$$

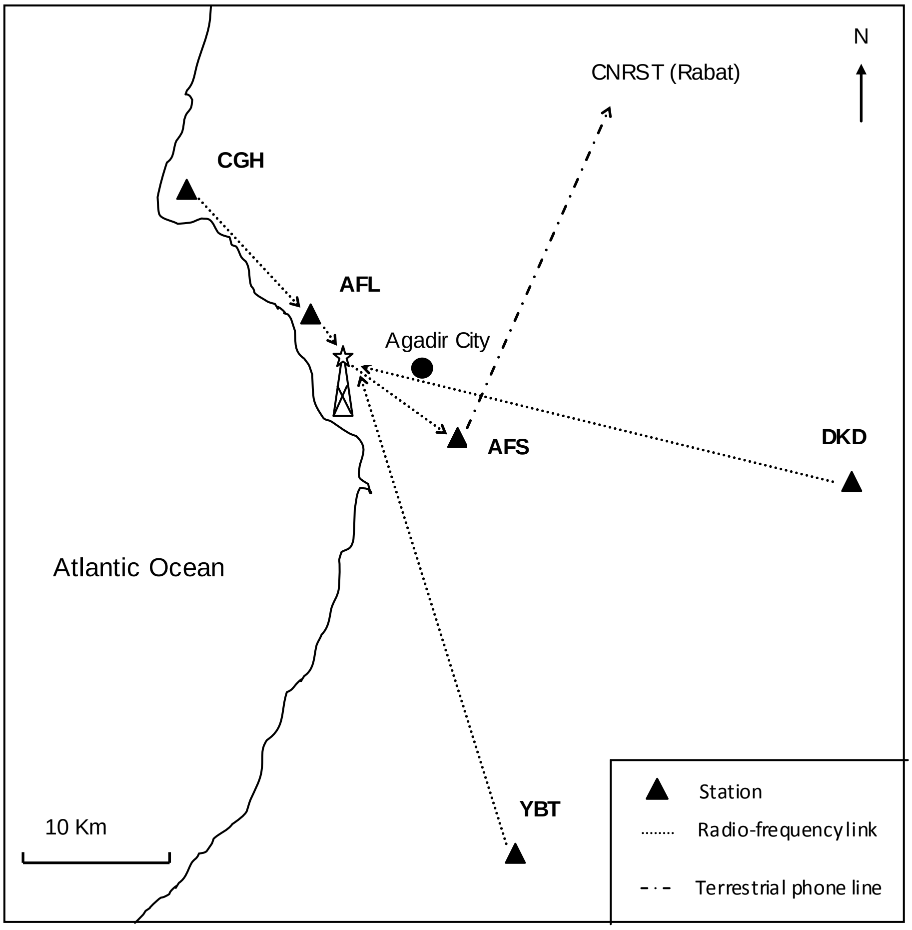

Fig 1. Station location map of the local seismic network of Agadir

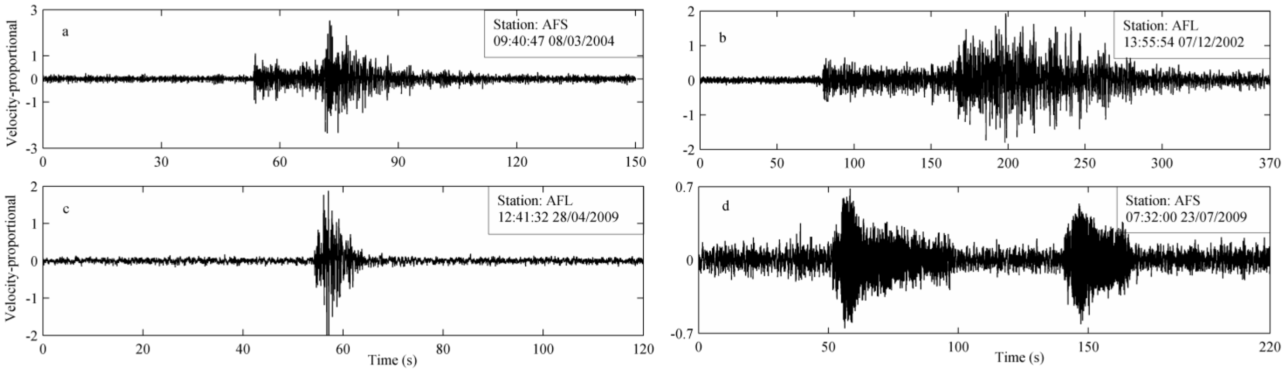

Fig. 2.Vertical component seismogram of four seismic events generated by different sources and recorded by the seismic local network of Agadir. (a) local earthquake, (b) regional earthquake, (c) quarry blast, (d) machinery.

Hour

Day time event distribution reveals that quarry explosions are generally launched between 11:00 a.m. and 02:00 p.m and between 05:00 p.m and 06:00 p.m GMT. Beyond this time intervals, explosion are absent and the seismicity pattern should not be affected by this type of event. Parameter hour *H*can be computed as follows:

$$H = hour+minute/60+second/3600$$

Spectral centroid

The Spectral centroidS_c indicates the barycenter of the event spectrum. This measure is obtained by computing the "center of gravity" using the normalized amplitude of FFT envelope weighted by its corresponding frequencies.

$$S_c = \frac{\sum_{i=1}^{N} f(i)e(i)}{\sum_{i=1}^{N} e(i)}$$

e(i) represents the amplitude of theFFT envelop of the bin number *i*, and *f(i)* represents its frequency.

Spectral length

Spectral lengthS_l of each event is estimated by applyingthresholds on its FFT envelop.

$$S_l=f_n-f_0$$

wheref_0 and f_n are the first and last selected frequency bins.

Skewness

Skewness is used here to characterize the degree of symmetry or asymmetry of an event signal around its mean. *S*for a roughly symmetrical signal is near zero. *S* of a real signal is given by:

$$S = \frac{\frac{1}{N}\sum_{i=1}^{N}(\mathrm{z(i)} - \bar{z})^3}{\left(\frac{1}{N}\sum_{i=1}^{N}(\mathrm{z(i)} - \bar{z})^2\right)^{3/2}}$$

N is the number of sample in the event signal zand $\bar{z}$ its mean.

3. CLASSIFIER STRUCTURE

3.1 MLP Neural network

Artificial neural network [23] involves an arrangement of many simple processing elements, called nodes or neurons, interconnected through weighted connection to implementcomplex non-linear mappings between inputs and outputs. Numerous types of neural networks exist in the literature. In this study, the most commonly used neural network, called supervised learning multilayer perceptron neural network, is consider. As its name indicates, it is composed of several layers. These are separated into input layer where the extracted features are received, output layer where the network indicates the predicted class, and hidden layers between the input and output layers. The input of each neuron in the hidden and output layers come from the outputs of the neurons in the precedent layers andfrom a constant input called the bias. Each neuron is characterized by a vector of weights that multiply its input, and by an activation function that calculates the output of the neuron from the weighed sum of its inputs.Theweights, along with the network architecture, store the knowledge of theneural network. An example of a MLP neural network with two hidden layers is shown in figure 3.

In order to use the MLP, it should be trained on a pair input/output set of data to learn to associate the inputs with the corresponding outputs. During the training process, each example in the training set is presented to the neural network, and a learning algorithm modifies the weights of the network in order to minimize the error between the desired and observed outputs. Mean squared error (MSE) is a commonly used metric to evaluate the difference. Each evaluation of all examples in the training set is called an epoch. In fact, the goal of the network training is not to learn an exact representation of the training data itself, but rather to build a model of the process which generates the data. The generalization capability of the neural network can be estimated by predicting the class of events that are not seen during training. Therefore, after training, the performance and generalization ability of the network has to be evaluated on an independent set of input/output examples, called data test set.

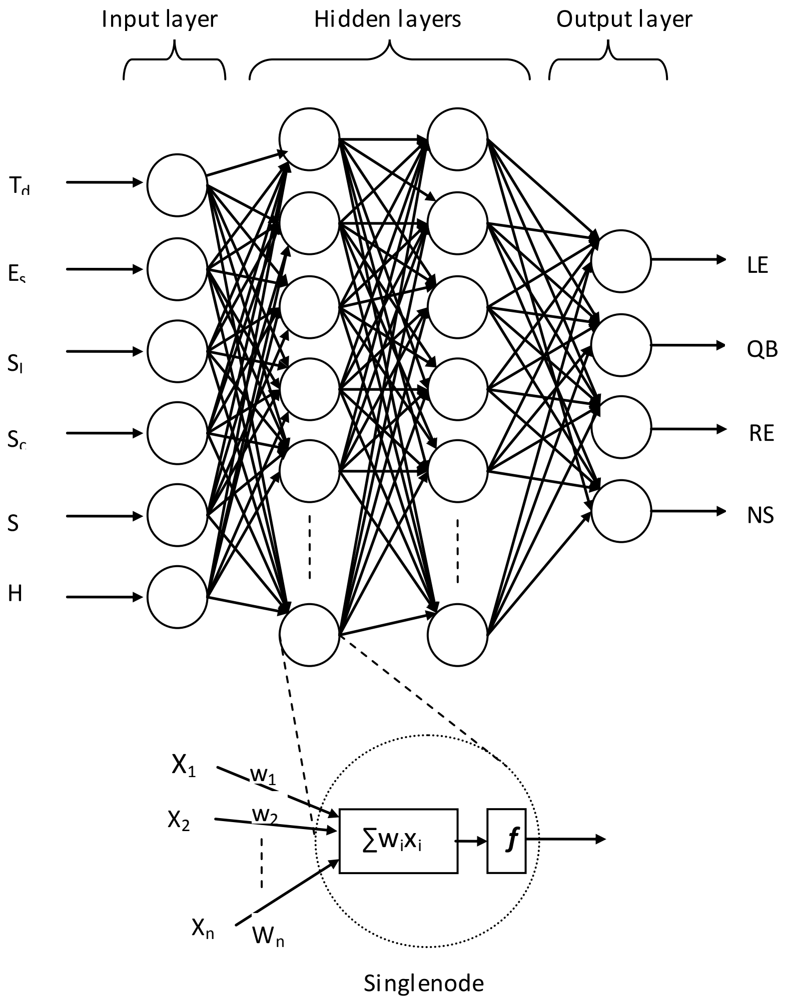

Fig. 3. Architecture of the multilayer perceptron neural network used to classify seismic events.

3.2 Neural network structure

The crucial and difficult task in designing a neural network is to determine an optimal topology that achieves the best results. Indeed, the choice of the network architecture and the activation functions as well as the learning algorithm has a significant impact on the network performance, its generalization skills and its training duration. For example, the number of connections in a neural network reflects its ability to store information. Therefore, if a network does not possess enough connections, the training algorithm may never converge and the network may not have enough flexibility to capture the nonlinearities in the data. Conversely, too many connections (hidden neuron) can lead to an excessively long training time and, worse, to overfitting. When overfitting occurs, the network will start to model random noise in the data. As result, the network has learnt the training data extremely well, but its generalization capability for unseen data is poor. This highlights the need to optimize the complexity of the model in order to achieve the best generalization.

In a classification problem, the number of neurons in the input layer is determined by the number of features that represent the signal. The number of output nodes is fixed by the number of classes; one output node per class. In this manner, four output neurones were used to represent the four classes as follows: LE-1000, NS-0100, QB-0010 and RE-0001. The appropriate number of hidden layers and the number of neurons required in each hidden layer as well as the activation functions remains an open question and problem complexity depending. They should be chosen in order to improve the classifier performance and generalization.

It has been established that, for nearly all problems, one hidden layer with sigmoid activation functions and enough neurons is sufficient to represent any functional relationship between inputs and outputs [24]. However, although it is capable of modeling such non linear relationships, it may require too many hidden neurons.

There exist several automatic methods of selecting the optimal architecture of a neural network such as genetic algorithms and adaptive processes [25][26][27]. In this study, a trial and error process was conducted.

3.3 Performance evaluation

The performance and generalization capability of a trained neural network should be generally tested using a set of data that are not used during training. One issue that is usually faced when using supervised neural networks is how to split the available data into a separated training set and test set. In practice, the data can be divided depending on its quantity. Sufficiently large data set can be divided into three mutually disjoint sets using the hold out method [28].These sets are: training set, validation set and test set. The training set is used to adjust the weights of the ANN during its training process. The validation set is used to stop training and avoid overfitting the data. The test set is used to evaluate the final classification performance. When the amountof available data is limited, it may not be sufficient to be separated into three independent sets to train and test the network. In this situation, applying cross-validation and especially k-fold cross-validation technique [28][29] is considered more accurate and reliable. This technique is commonly used when comparing the performance of two or more supervised neural network models. The k-fold cross-validation consists in partitioning randomly the available data into k disjoint subsets (folds) of equal (or approximately equal) size. K iterations of training and testing are then performed. Within each iteration, a different subset of the data is used for testing whereas the remaining k-1 subsets are used for learning. The k test results are then averaged, and the model with the best outcome is chosen. The samples should be distributed among the k folds in a way that each fold contains examples from all classes.

Given a set S of m examples, a single run of k-fold cross validation proceeds as follows:

1. Arrange the examples in a random order.
2. Divide S into k disjoint equal-sized folds (approximately m/k examples each.): $S_1, \ldots, S_k$

For i= 1 to k do

- Train the classier using all the examples that do not belong to Fold i.
- Test the classier on all the examples in Fold *i*.
- Compute the confusion matrix of the test *i*

3. Return the total confusion matrix

The network training process used the 'early stopping' mechanism [29] to avoid overtraining and to reduce training time. In this technique, in each iteration of k-fold cross-validation,a potion of the training data is randomly selected to be used as a validation set.The idea is to evaluate the training and validation error simultaneously and stop the training process when the validation error starts increasing. The training error is obtained using the training set and the validation error is calculated using the validation set.

Since the training process is based on optimizing the error function between the target and output of a neural network, finding the global minimum is not guaranteed as the error surface can include many local minima in which the algorithm can become stuck. To improve the result, the network should be trained several times with different initializations of weights and biases. The solution with the lowest error is then chosen.

4. RESULTS AND DISCUSSION

To choose the best classifier topology and evaluate its performance, a data set of 343 event seismograms was chosen and classified by seismic analysts. These seismograms are distributed among the four classes as depicted in table 1.

Table 1.Distribution of seismograms in the four classes

class	LE	NS	QB	RE
Number of event	83	115	100	45

After data processing, the six input features were extracted.Thirteen MLP neural networks of different number of neuron in hidden layer were created.Each configuration wastrained and then tested on independent data sets.To obtain more reliable results, the k-fold cross validation method was used to estimate each classifier generalization capability. To do so, the data were randomly split into five approximately equal partitions (k=5). In each partition, the classifier was tested after being trained using the four remaining partitions. Using this methodology, the process of training is repeated 5 times, each time using a different test set chosen from the 5 partitions until all the five sets have been used. To avoid overtraining, the early stopping strategy was used. After each epoch, the network is tested on a validation data set. The error produced by the network determines wether the training process should be stopped or continued. Figure 6 shows the training and validation errors for the first iteration of the 5-fold cross-validation (i=1). As displayed in figure 6, the training error decreases as training continues, and the validation error normally deceases during the initial phase of training. However, when the network begins to overfit the data, the error on the validation set will typically begin to increase. The training process is stopped when the validation error reaches its minimum.

For each architecture, the results of the 5 tests are averaged, and the model with the best outcome is selected. The classification result of each test is summarized in a confusion matrix that shows the number of correctly classified and misclassified events of each class. The output of the cross-validation is the sum of the five confusion matrices. Thus it indicates the finale performance of the classier on the five test sets. The average recognition percentages (correctly classified signals/total) obtained by each configuration are represented in figure 5. It can be seen that performance increases very quickly with the number of neurons in the hidden layer. The performance becomes stable thereafter between 4 and 7 neurons. But, when the number of hidden neurons exceeds 7, the performance starts to change significantly and becomes unstable. In accordance with these results, the model with five hidden neurons was selected to be the best topology. This is because of the best performance achieved with a small standard deviation. Figure 4 illustrates the results of the selected model in more details. Each confusion matrix shows in the columns the target classes and in the rows, the predicted classes. The diagonal indicates agreement and the other cells indicate the misclassified events.

To avoid converging to a local minimum, the network was trained 10 times with random initializations of weights and biases. The solution with the lowest error was then chosen.

The performance of the classifier was evaluated by the computation of the following parameters:

Sensitivity (Recall): It measures the ratio between the number of correctly classified signals of a class and the total number of signals of that class. Therefore, it measures the ability of the classifier to recognize signals of a particular class.

Specificity: It is the ratio between the number of signals that were classified as not belonging to a class and the total signals that did not belong to that class, and thus it measures the ability to recognize signals that are not of a particular class.

LE	15	1	0	2
NS	0	22	0	0
QB	1	0	19	0
RE	0	0	1	7
	LE	NS	QB	RE

Test 1

LE	13	0	3	0
NS	1	23	0	0
QB	3	0	17	0
RE	0	0	0	9
	LE	NS	QB	RE

Test 2

LE	16	0	1	0
NS	0	22	0	0
QB	1	0	19	0
RE	0	1	0	9
	LE	NS	QB	RE

Test 3

LE	15	0	2	0
NS	1	23	0	0
QB	0	0	18	0
RE	0	0	0	9
	LE	NS	QB	RE

Test 4

LE	16	1	0	0
NS	0	22	0	0
QB	1	0	19	0
RE	0	0	1	9
	LE	NS	QB	RE

Test 5

LE	75	2	6	2
NS	2	112	0	0
QB	6	0	92	0
RE	0	1	2	43
	LE	NS	QB	RE

Total

Fig. 4. Confusion matrix of the five tests for the chosen five hidden neurons MLP. The columns correspond to the target classes, and the rows correspond to the predicted classes

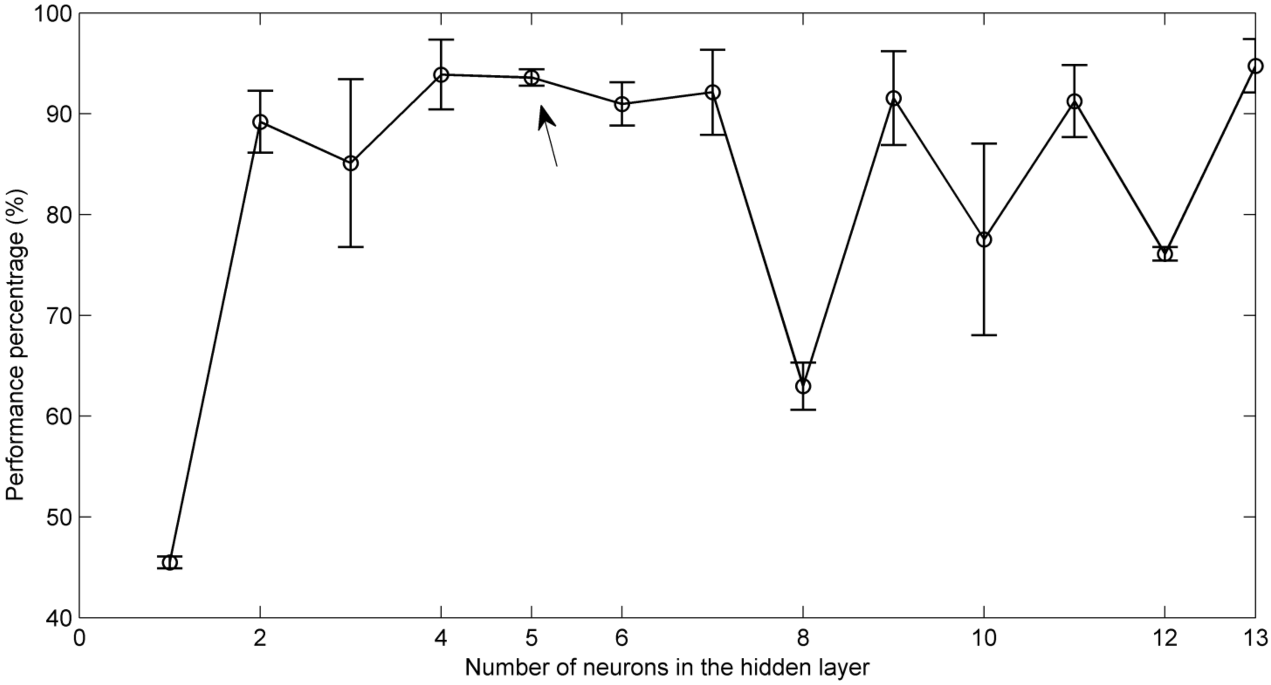

Fig. 5. Mean performance and standard deviation as a function of the hidden unit numbers. The arrow indicates the result of the selected 5 hidden-neuron layer.

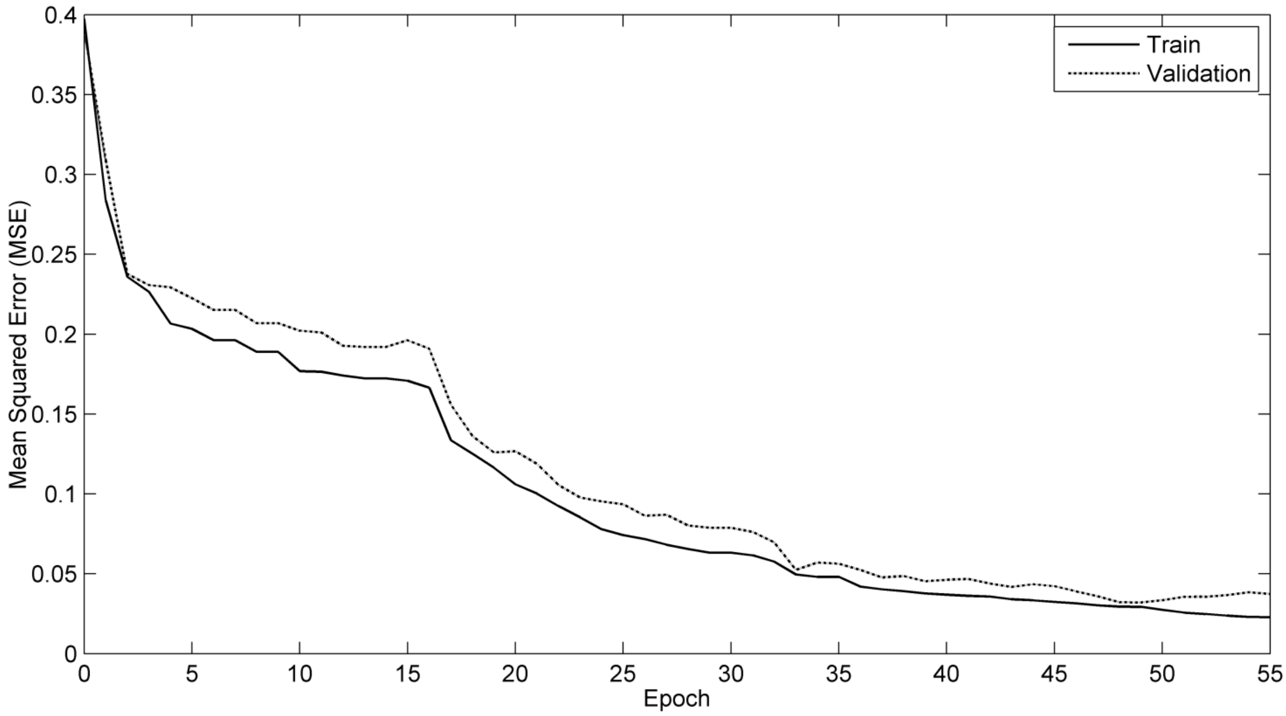

Fig. 6. Training and validation errors of the best selected classifier (5 neurons in the hidden layer)in the first test run as a function of the training epoch number.The training process is stopped when the validation error starts to increase (epoch= 50).

Table 2.Performance evaluation of the classifier for the four classes.

	Sensitivity (%)	Specificity (%)	Precision (%)	Accuracy (%)	Error (%)
LE	90,36	96,15	88,24	94.75	5.25
NS	97,39	99,12	98,25	98.54	1.46
QB	92,00	84,95	93,88	95.92	4.08
RE	95,55	98,99	93,48	98.54	1.46

Accuracy (Exactitude): It measures the ratio of correctly classified signals to the total signals.
Error: It measures the ratio between the incorrectly classified signals to the total signals.
*Precision:*It measures the ratio between the number of correctly classified signals of a class and the total number of signals assigned to that class.Therefore, it represents the capability of the classifier to not include signals of other classes in the considered class.
These parameters were extracted from the final confusion matrix (figure 4) and are presented in Table 2 for each class.
As shown in Table 2, the specificity of the classifier is better than its sensitivity except for the class QB. The class NS posses the highest value of sensitivity and specificity as well as precision. The classifier is more able to recognize the signals of the classes RE and NS than the classes LE and QB. The accuracy of the classifier is generally good.
The performance of the MLP neural network is found to be satisfactory and we think that this system may be made more accurately by increasing the variety and number of input features.

5. CONCLUSION

In this work, the performance of a MLP neural network for classifying seismic signals recorded by the local seismic network of Agadir was examined. Based on a set of signals used for evaluation, it was demonstrated that the MLP classifier is able to identify the class of each signal with an acceptable accuracy. However, the classifier is considered as a black box for seismologists and the results cannot be interpreted. Moreover, due to the small size of the data used, the generalization capability of the classifier should not be trusted. Thus, the proposed method is not yet being employed in the monitoring system. Future work is oriented first at performing a deep analysis of signals of each class in order to extract more efficient discriminant features. A second concern is to design 'an online' classification system that is able to integrate expert knowledge and improve the classification results.

6. REFERENCES

[1] M. Hedlin, J.B. Minster, J.A. Orcutt. An automatic means to discriminate between earthquakes and quarry blasts. Bull. Seism. Soc. Am. 80, 2143–2160, 1990.

[2] Y. Gitterman and A. Shapira. Spectral discrimination of underwater explosions. Israel Journal of Earth Science 42, 37–44, 1993.

[3] Y. Gitterman, V. Pinky and A. Shapira. Spectral classification methods in monitoring small local events by the Israel seismic network. J. Seismology 2, 237–256, 1998.

[4] K. Koch and D. Fah. Identification of earthquakes and explosions using amplitude ratios: theVogtland area revisited. Pure and Applied Geophysics 159,735–757, 2002.

[5] P. B. Allmann, P. M. Shearer and E. Hauksson. Spectral discrimination between quarry blasts and earthquakes in Southern California. Bull. Seism. Soc. Am., Vol: 98, Issue: 4, Pages: 2073-2079, 2008.

[6] S.A. Dahy and G.H. Hassib. Spectral discrimination between quarry blasts and micro earthquakes in southern Egypt. Research Journal of Earth Sciences 2 (1), 1–7, 2010.

[7] Kushnir, A. F., V. M. Lapshin, V. I. Pinsky, and J. Fyen (1990). Statistically optimal event detection using small array data, Bull. Seism. Soc. Am. 80, 1934–1950.

[8] J. Wuster. Discrimination of chemical explosions and earthquakes in central Europe—a case study. Bull. Seism. Soc. Am. 83, 1184–121, 1993.

[9] A. F. Kushnir, E. V. Troitsky, L. M. Haikin, and A. Dainty. Statistical classification approach to discrimination between weak earthquakes and quarry blasts recorded by the Israel Seismic Network, Phys. Earth Planet. Int. 113, 161–182, 1999.

[10] D. B. Harris. A waveform correlation method for identifying quarry explosions. Bull. Seism. Soc. Am. 81, 2395–2418, 1991.

[11] P. Gendron, J.Ebel, and D. Manolakis (2000).Rapid joint detection and classification with wavelet bases via bayes Theorem, Bull. Seism. Soc. Am. 90, 764–774.

[12] S. Falsaperla, S. Graziani, G. Nunnari, S. Spampinato. Automatic classification of volcanic earthquakes by using multi-layered neural networks. Natural Hazards 13, 205–228, 1996.

[13] M. Musil and A. Plesinger. Discrimination between local micro earthquakes and quarry blasts by multi-layer perceptrons and Kohonen maps. Bull. Seism. Soc. Am. 86 (4), 1077–1090, 1996.

[14] S. Muller, P. Garda, J.D Muller and Y. Cansi. Seismic events discrimination by neuro-fuzzy catalogue features. Physics and Chemistry of the Earth 24 (3), 201–206, 1999.

[15] F.U. Dowla, S.R. Taylor and R.W Anderson. Seismic discrimination with artificial neural networks: preliminary results with regional spectral data. Bull. Seism. Soc. Am. 80, 1346–1373, 1990.

[16] T. Tiira. Detecting teleseismic events using artificial neural networks. Computers and Geosciences 25, 929–939, 1999.

[17] R.D. Jenkins and T.S. Sereno. Calibration of regional S/P amplitude-ratio discriminants. Pure and Applied Geophysics 158, 1279–1300, 2001.

[18] A. Ursino, H. Langer, L. Scarfı, G. Di Grazia and S. Gresta. Discrimination of quarry blasts from tectonic microearthquakes in the Hyblean Plateau (south- eastern Sicily). Annali di Geofisica 44, 703–722, 2001.

[19] E. Del Pezzo, A. Esposito, F. Giudicepietro, M. Marinaro, M. Martini, S. Scarpetta. Discrimination of earthquakes and underwater explosions using neural networks. Bull. Seism. Soc. Am. 93, 215–223, 2003.

[20] S. Scarpetta, F. Giudicepietro, E.C. Ezin, S. Petrosino, E. Del Pezzo, M. Martini, M. Marinaro. Automatic classification of seismic signals at Mt. Vesuvius volcano, Italy, using neural networks. Bull. Seism. Soc. Am. 95, 185–196, 2005.

[21] E. Yıldırım, A. Gulbag, G. Horasan and E. Dogan: Discrimination of quarry blasts and earthquakes in the vicinity of Istanbul using soft computing techniques. Computers & Geosciences, 01-09, 2010.

[22] E. Akhouayri, D. Agliz, M. Fadel, and A. Ait Ouahman. Automatic detection and indexation of seismic events. AMSE Periodicals, Advances in Modeling and Analysis, S. C, Vol. 56, 59-67, 2001.

[23] M. BishopChristopher, Neural networks for Pattern recognition. Oxford University, New York, 1995. 116-161pp

[24] K.Hornick, 1991, Approximation capabilities of multilayer feedforward neural network. Neural network, 4(2), pp. 251-257

[25] B. Zhang and H. muhlenbein. Evolving optimal neural networks using genetic algorithms with occam’s Razor. Complex system 7(1993) 199-220

[26] V. K. Devabhaktuni, M. Yagoub, and Q. J. Zhang, “A robust algorithm for automatic development of neural-network models for microwave applications,” IEEE Trans. Microwave Theory Tech., vol. 49, pp.

[27] T. Y. Kwok and D. Y. Yeung, “Constructive algorithms for structure learning in feedforward neural networks for regression problems,” IEEE Trans. Neural Networks, vol. 8, pp. 630–645, May 1997

[28] Z. Reitermanova, Data Splitting. WDS’10 Proceedings of contributed papers, part I, 31-36, 2010

[29] T. Hastie, R. Tibshirani and J. Friedman. The Elements of Statistical Learning Data Mining, Inference, and Prediction. Springer Series in Statistics, second edition, 2008.